REMARKS

Claims 1-3, 10-12, 19-34 and 36-48 are pending in the present application. Claims 1-3, 10 and 36 were amended in this response. No new matter has been introduced as a result of the amendments.

Claim 36 was rejected under 35 U.S.C. §112, second paragraph since the phrase "said first segment checksum" lacked antecedent basis. In light of the present amendment, Applicants submit the informality has been addressed. Withdrawal of the rejection is earnestly requested.

Claims 1-3, 10, 22-33, 37, 40, 43 and 46 were rejected under 35 U.S.C. §103(a) as being unpatentable over the publication "Data Communications, Computer Networks and Open Systems" by *Halsall* (herein after "*Halsall*") in view of *Frezza* (US Patent 4,982,430).

Claims 11, 12, 19-21, 34, 36, 38, 39, 41, 42, 44, 45, 47 and 48 were rejected under 35 U.S.C. §103(a) as being unpatentable over the publication "Data Communications, Computer Networks and Open Systems" by *Halsall* (herein after "*Halsall*") in view of *Frezza* (US Patent 4,982,430) and further in view of Mattison (US Patent 5,778,070). Applicants respectfully traverse these rejections. Favorable reconsideration is earnestly requested.

Specifically, the cited art, alone or in combination, does not teach a commutative operation which forms said first commutative checksum by operating on the first segment checksums (note: plural) as recited in claim 1, and similarly recited in claims 2-3 and 10.

As argued previously, *Halsall* teaches a method for detecting errors occurred during a transmission of a bit stream, whereby a commutative checksum is used to identify the errors in the transmission of the data. Under *Halsall*, an odd or even parity bit is assigned to each data segment (row/column), which corresponds to the first segment checksums under the present claims ("forming a first segment checksum for each said data segment"). As a result, *Halsall* obtains a transverse (row) parity bit vector and a longitudinal (column) parity bit vector (see page 129, figure 3.15). A parity checksum and a cyclic redundancy checksum method is also used, whereby the parity checksum methods are best suited to applications in which single-bit errors are present, whereas the cyclic redundancy checksum method is best suited to applications in which bursts of errors occur.

The Office Action implies that the block comprising the column parity bits in *Halsall* teaches the first commutative checksum corresponding to the present claims. However, this is

incorrect, because the column parity bits vector is not formed by a commutative operation - only the individual first segment checksums are subject to a commutative operation (i.e., XOR) (see page 127, 3.4.1, 4th paragraph, page 129). Unlike the disclosure in *Halsall*, the present claims recite checking a first commutative checksum for digital data grouped into a number of data segments, in which flow control for the individual data segments is not required. In other words, Halsall would need a mechanism or process by which the resulting vector is subject to a commutative operation. As explained above, it does not.

Halsall teaches a parity checksum and a cyclic redundancy checksum method, whereby the parity checksum methods are best suited to applications in which random single-bit errors are present, whereas the cyclic redundancy checksum method is best suited to applications In which bursts of errors occur. *Halsall* further teaches that both of the aforementioned methods have a disadvantageous effect, in that they are only able to identify errors under very specific conditions. Using parity checksum, only two bit errors can be detected in a character, and only if no two bit errors occur in the same column at the same time (page 129, lines 12 *et al.*). Using cyclic redundancy, the length of an error burst can only be determined if the last erroneous bit in a burst and the first erroneous bit in the following burst are separated by *B* or more correct bits, where *B* is the length of the error burst (page 130, sec. 3.4.3, lines 6 *et al.*).

As such, the disclosure in *Halsall* would not be able to deal with the problems addressed in the present claims in protecting digital data against unauthorized modification. The data integrity is protected under the present claims if the receiver of the message can detect every error vis-à-vis the commutated checksums. Through this configuration, digital signatures like hash values can be used, which are able to detect errors starting at one bit errors under all conditions. Such a configuration is neither taught nor suggested in *Halsall*.

Furthermore, *Frezza* does not solve the deficiencies of *Halsall*, discussed above. *Frezza* discloses a communication system where users are allowed to securely download data from a remote site (col. 1, line 55-col. 2, line 9). During a download process, booter data is downloaded to a user terminal from a booter 14 to establish a subscriber identity (col. 4, lines 18-46). Subsequently, a checksum operation is performed on the booter data to validate the user, and an encrypted communication link is established with the terminal to the network control center (NCC) by transmitting the encrypted checksum (col. 5, line 39 - col. 6, line 19). Thus, Frezza

also fails to teach or suggest cryptographically protecting a commutative checksum that is formed by a commutative operation on the first segment checksums.

In light of the above, Applicants respectfully submit that independent claims 1-3 and 10-12 of the present application, as well as claims 19-34 and 36-48 which respectively depend therefrom, are both novel and non-obvious over the art of record. Accordingly, Applicants respectfully request that a timely Notice of Allowance be issued in this case. If any additional fees are due in connection with this application as a whole, the Examiner is authorized to deduct said fees from Deposit Account No.: 02-1818. If such a deduction is made, please indicate the attorney docket number (0112740-466) on the account statement.

Respectfully submitted,

BELL, BOYD & LLOYD LLC

BY 

Peter Zura
Reg. No. 48,196
Customer No.: 29177
Phone: (312) 807-4208

Dated: April 11, 2006